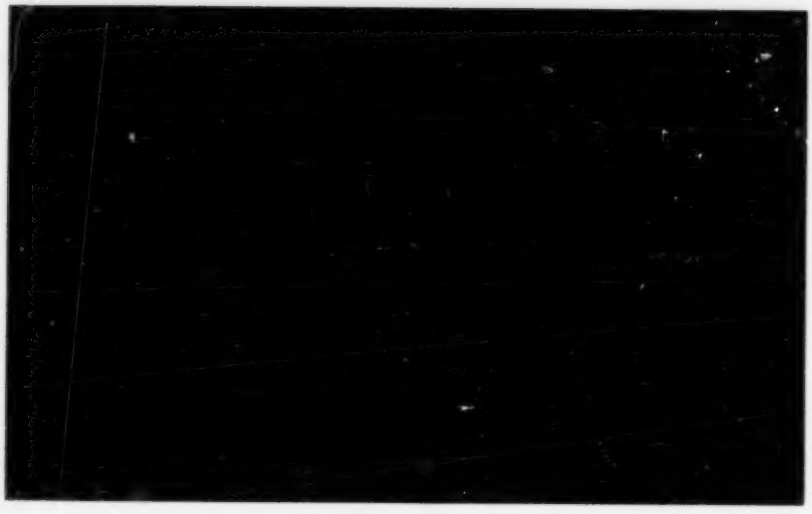
CENTRE DE RECHERCHE SUR LES TRANSPORTS
(C.R.T.)
CENTRE FOR RESEARCH ON TRANSPORTATION

UNIVERSITÉ DE MONTRÉAL
ÉCOLE DES HAUTES ÉTUDES COMMERCIALES
ÉCOLE POLYTECHNIQUE



CENTRE DE RECHERCHE SUR LES TRANSPORTS
UNIVERSITÉ DE MONTRÉAL
C.P. 6128, SUCCURSALE A,
MONTRÉAL
CANADA H3C 3J7

TÉLÉPHONE: (514) 343-7575
TÉLEX: (21) 05-561325
EMAIL: crt@crt.umontreal.ca
TÉLÉCOPIEUR (FAX): (514) 343-7121

CRT-768

Real-Time Scheduling on a Transit Bus Route

by
Yihua Li[1]
Jean-Marc Rousseau[2]
Fujun Wu[3]

[1] Centre de recherche sur les transports, Université de Montréal, C.P. 6128, succursale A, Montréal, Canada H3C 3J7.
[2] Centre de recherche sur les transports and Département d'informatique et de recherche opérationnelle, Université de Montréal, C.P. 6128, succursale A, Montréal, Canada H3C 3J7.
[3] Shanghai Institute of Mechanical Engineering, Shanghai 200093, People's Republic of China.

Centre de recherche sur les transports — Publication #768
Département d'informatique et de recherche opérationnelle — Publication #771

April 1991

91-04845/

Dépôt légal – 2ème trimestre 1991

ABSTRACT

We consider a scheduling problem in real-time on a single transit bus route where buses may be express and stop only at a few places or turn before the terminus in order to respect the schedule. This paper describes methods based on heuristic procedures designed to provide quick solutions to the real-time scheduling problem. The quality of the results are evaluated on a test problem using data from Shanghai transit company.

Keywords: Real-time, transit scheduling, stop skipping patterns.

RÉSUMÉ

Nous étudions le problème de dresser en temps réel l'horaire d'une ligne d'autobus unique, quand certains autobus sont des express qui n'arrêtent qu'à certains arrêts, ou qui peuvent revenir en sens inverse avant d'atteindre le terminus. Cet article décrit les méthodes basées sur des procédures heuristiques visant à l'obtention rapide de solutions en temps réel. La qualité des résultats produits par cette approche est évaluée sur un problème test bâti à partir des données de la compagnie de transport en commun de Shanghai.

Mots-clés: Temps réel, horaires de transport en commun, saut d'arrêts.

1. INTRODUCTION

1.1 General Description

We consider the scheduling problem for the operation of transit bus and trolley lines in Shanghai, China. The problem is different from the one encountered in western countries because the number of buses is not sufficient to satisfy the transit demand. Moreover because of congestion on the street, bus running times vary significantly which would necessitates large layover time to respect any planned schedule.

However, because the supply of buses is insufficient, it is not reasonable to plan for large layover at the terminus of bus lines. It would be wasting transit capacity. In addition, to increase further the capacity of bus lines, all buses on a line do not always stop at every bus stop. Some buses are express and stop at only a few places. Some buses will not restart from the terminus but at other points on the line to provide capacity to passengers waiting at intermediate points. Some buses will make short turns, meaning that they do not continue to the terminus but turn back before the end of the line. These special trips have to be made at certain times of the day to respond to the travel demand.

Because of the congestion and lack of bus capacity, it is not really possible to efficiently plan the bus schedule in advance. There is thus a need for a system to control the schedule in real time. The scheduling problem in this context can be decomposed into two categories: the operational planning of the schedule and real time control of it.

In operational planning, the problems are the following:

a) we must decide how many vehicles to allocate to each bus line in order to equilibrate service among bus lines given that there is a shortage of buses;

b) for each bus line, we must decide which patterns of services (series of stops) should be performed according to the time of day and predicted travel demands;

c) we must prepare an approximate bus driver schedule.

In real time control, the problem is to decide which pattern of services a bus leaving the terminus must perform given the operational plan, the previous decision taken and the anticipated decision for the future.

In order to achieve that objective, it will be necessary to estimate (if no real time data is available) the location of buses or trolleys on the line. To test the various strategies and validity of the model, a simulation of a line operation will have to be developed. This will be the subject of a future paper. In another paper, methods to generate a pre-determined bus schedule have been developed[11]. In this paper we present our approach for the real-time control problems.

1.2 Literature Review

Routing and scheduling of transit vehicles and crews has been a subject of interest and literature on the problem has been growing rapidly over the last decade. A general survey of routing and scheduling of vehicle and crews was made by Bodin et al.[2] and Rousseau[14] respectively. In his survey, Rousseau classified problems according to the various levels of planning. Strategic planning is related to long term commitments such as subway and light rail development and also modifications to the bus route structure. Operational planning problems are concerned with all the scheduling-related activity for which Scott[16] gave an excellent survey of the development of approaches. Ceder[4] presented an optimal design of transit short-turn trips to minimize fleet size where trips are initiated beyond the route departure point, or terminated before the route arrival point, or both. Several bus scheduling software packages are available, some of these integrated into a crew scheduling software (see Wren[17], Rousseau[13], Daduna and Wren[6], and Desrochers[7]). Most

of this research focuses on planning or pre-scheduled operational problems. The real-time scheduling problem in public transit, however, has received little attention. Breusegem and Campion[3] developed a linear model to carry out traffic control on a looped metro line. Cury et al.[5] presented a system for real time control of the underground railway system of Sáo Paulo. Cury et al. established an analytical model to describe the flow of passengers boarding and leaving each train, the motion of trains, and the quality of performance. The model also minimize the total deviations from a preestablished schedule. A decentralized control methodology was used by them for the "on line" control of train traffic in underground railway systems. A number of transit queuing and simulation models have been developed by Bly and Jackson[1] Koffman[10] and Ryan[15] to evaluate the impact of different dispatching alternatives.

In this paper we are concerned with a different problem. Due to heavy traffic congestion and a shortage of bus capacity, delays of bus arrivals at the terminus occur frequently with the delay time varying from 5 to 30 minutes where the service headway are relative short (2-4 minutes on the average). Occasionally, therefore, not only the bus departure time must be adjusted but the bus running time as well, by skipping a number of stops in order to reestablish the desired headway. By doing this, we have to take corrective strategies to minimize total passenger waiting time at stops along a route and make up the time on a preestablished schedule. This is the subject of our paper.

2. FORMULATION OF THE PROBLEM

2.1 Introduction

In real time scheduling, we must determine the bus departure time at the terminus and the stops to be skipped. Suppose there are N_s stops along a line, and N_b buses departing in a given time period. Then, the variables are: N_b dispatching variables for bus departure times at terminus (Td_{i1}=1,...,N_b), and N_b x N_s stopping variables (S_{ij} , i=1,...,N_b; j=1,...,N_s) while Td_{i1} is a real variable and S_{ij} is a 0-1 variable indicating if the i^{th} departing bus stops at stop j. The optimization of bus dispatching is to determine the values of Td_{i1} and S_{ij} in order to provide the best service to the public.

The passenger waiting time and boarding/alighting times are very difficult to represent in linear functions. Another difficulty is the number of 0-1 variables. Because of these difficulties, we have to select one of the following strategies:

1) Simplify the model so that it can be solved easily (by linear programming for example).

2) Use an accurate model but solve the problem using approximate methods.

Alternative 1 has the advantage that we can solve the problem optimally. However, since the model is not accurate enough, this alternative can not represent the real problem correctly and the solution obtained would not be useful to solve the problem in real time.

Alternative 2 is more practical. In real life dispatching, the main consideration is how to correctly evaluate the current situations and give quick responses. We will develop such an approach.

2.2 Problem Approach

We assume that the number of buses which can interact in the dispatching process is limited by a constant N_b. We assume also that the N_b^{th} bus will be on schedule and will not skip any stop. This is a realistic assumption, because we only

need to control a limited number of dispatching buses to eventually get back on schedule. So we choose the following approach:

- Solve the dispatching problem for the next n buses, which means finding
  - the departure times (except for the last bus)
  - the pattern of stops (except for the last bus);
- Use results from the model to dispatch only the next bus arriving at terminus
- Re-solve the problem for every bus arriving at the terminus.

2.3 Assumptions of the problem

The problem of real-time scheduling could be very complex. For simplicity of problem solving and presentation, we will make a set of assumptions.

(1) Passengers at a stop may take several bus routes to their destination. We assume that the proportion of passenger taking each bus route can be calculated from a ridership survey.

(2) No allowance is made for any bus to pass other buses. This is certainly true for trolley lines. This assumption should eventually be relaxed.

(3) We assume that delay for any passenger is increased by a penalty factor d if a bus of his desired line skips his stop.

(4) Passengers will wait at their stop until a bus arrives (nobody leaves the system without taking the bus).

(5) We assume that the last bus in the dispatching problem leaves on time and does not skip any stops.

(6) We are working on a circular line that starts at stop 1 and finishes at stop 1. That is, there is only one control point (terminus) on the line at stop 1. This assumption does not limit the generality of the model.

2.4 Mathematical Model

The problem can be formulated as follows:

Define the following constants:

N_b -- number of buses concerned in the problem

N_s -- number of stops on a route (stop 1 and stop N_s both correspond to the same terminus)

a_{jk}-- the passenger travelling rate from stop j to stop k

$a_{j\circ}$-- the passenger arriving rate at stop j, that is, $a_{j\circ} = \sum_{k>j} a_{jk}$ (1)

d -- associated penalty factor with passenger waiting while a bus skip their stop

t_a -- alighting time per passenger

t_b -- boarding time per passenger

t_{fix} -- time lost during acceleration and deceleration

T_{fix} -- the fixed scheduled departure time for the last bus considered in the problem

V_i -- the capacity of the i^{th} bus

Define the following variables:

Pb_{ijk}--the number of passengers boarding on bus i and travelling from stop j to stop k

Pw_{ijk}- the number of passengers waiting for the i^{th} bus(at the arrival time of the i^{th} bus at stop j) to travel from stop j to stop k, where the number of passengers (L_{jk}) left behind by the last bus before our scheduling period is defined as:

$$L_{jk} = Pw_{0jk} - Pb_{0jk} \tag{2}$$

S_{ij} -- the stopping variable for the ith bus arriving at stop j

$$S_{ij} = \begin{cases} 1 & \text{if bus } i \text{ stops at stop } j \\ 0 & \text{otherwise} \end{cases} \tag{3}$$

Ta_{ij} --the arriving time of bus i at stop j, where Ta_{i1} is the estimated time at which

the bus that will serve the i^{th} departure will be ready to load departing passengers

Td_{ij} --the departure time of bus i at stop j, where Td_{0j} is the departure time of the last bus before our scheduling period at stop j

Tt_{jk} --the bus travel time from stop j to stop k (from the departure time at stop j to arrival time at stop k)

Model:

The objective function selected is to minimize total passenger waiting time:

$$\textit{Minimize} \quad W = \sum_{i=1}^{N_b} \sum_{j=1}^{N_s-1} Tw_{ij} \qquad \textit{that is,} \tag{4}$$

$$\textit{Minimize } W = \sum_{i=1}^{N_b} \sum_{j=1}^{N_s-1} \left[\sum_{k>j} (Pw_{i-1,jk} - Pb_{i-1,jk}) + \frac{1}{2} a_j \cdot (Td_{ij} - Td_{i-1,j}) \right] (Td_{ij} - Td_{i-1,j}) \left[1 + d(1 - S_{ij})\right]$$
$$+ \sum_{j=1}^{N_s-1} \sum_{k>j} (Pw_{N_b j k} - Pb_{N_b j k}) \frac{1}{N_b} \sum_{i=1}^{N_b} (Td_{ij} - Td_{i-1,j}) \tag{5}$$

This function corresponds to the waiting time of passengers that are left behind by bus i-1, plus the waiting time of arriving passengers at stop j between the departure time of bus i-1 and arrival time of bus i. The waiting time is penalized by d if the i^{th} bus skips stop j. The second part of the function is a penalty for all passengers left behind by the last bus N_b.

This objective function is subject to the following:

i) Departure / arrival time constraints:

- The bus arrival time must be greater then or equal to its previous departure time plus bus travel time and plus the fixed lost time due to acceleration and deceleration:

$$Ta_{ij+1} \geq Td_{ij} + Tt_{jj+1} + t_{fix} \cdot S_{ij+1} \qquad i=1,\ldots,N_b; \quad j=1,\ldots,N_s-1 \tag{6}$$

- It is not allowed to pass other buses:

$$Ta_{i+1\,j} \geq Td_{ij} \qquad i=1,\ldots,N_b-1, \quad j=1,\ldots,N_s \tag{7}$$

- The bus departure time must be greater than or equal to its arrival time plus alighting and boarding time:

$$Td_{ij} \geq Ta_{ij} + t_a \sum_{q<j} Pb_{iqj} + t_b \sum_{k>j} Pb_{ijk} \quad i=1,...,N_b, j=1,...,N_s \tag{8}$$

- The departure time of the last bus of the scheduling period at the terminus must be on schedule, by assumption:

$$Td_{N_b 1} = T_{fix} \tag{9}$$

ii) Boarding constraints:

- The number of boarding passengers who can board an arriving bus is limited by: the remaining capacity on the bus and the pattern of stop, Let:

$$r = \frac{V_i - \sum_{k>j} \sum_{q<j} Pb_{iqk}}{\sum_{k>j} Pw_{ijk} S_{ik}} \qquad for\ S_{ij}=1 \tag{10}$$

then

$$Pb_{ijk} = \begin{cases} r \cdot Pw_{ijk} S_{ik} & if\ r<1\ and\ S_{ij}=1 \\ Pw_{ijk} S_{ij} S_{ik} & otherwise \end{cases} \tag{11}$$

while the number of passengers waiting for bus i at stop j for stop k is:

$$Pw_{ijk} = Pw_{i-1,jk} - Pb_{i-1,jk} + a_{jk} (Td_{i-1,j} - Td_{ij}) \tag{12}$$

where $i = 1, ..., N_b$; $j = 1, ..., N_s$ and $k = j+1, ..., N_s$.

Obviously, the model above is a 0-1 mixed non-linear programming problem with a quadratic objective function; it is too difficult to be solved exactly and we have to find some heuristic algorithms which can provide quick and satisfactory solutions.

3. INPUT DATA

3.1 Available Data

In this paper, we assume that the following data is available:

. the bus travel time between each pair of stops

. the number of passengers travelling from stop i to stop j

3.1.1 Bus travel time

Tt_{jk} depends on the road conditions and it usually belongs to truncated Gamma distribution $\Gamma(\alpha,\beta,c)$ (Ryan[9]), where the parameter β does not vary from stop to stop while α and c can be determined by observation for each pair of stops. Then bus journey time is calculated as:

$$Tj_{ijk} = t_{fix} + Tg_{ij} + Tb_{ij} + Tt_{jk} \qquad (13)$$

where Tj_{ijk} = the ith bus journey time from stop j to stop k.

t_{fix} = time lost during acceleration and deceleration.

Tg_{ij} = alighting time from bus i at stop j if the bus stops.

Tb_{ij} = boarding time on bus i at stop j if the bus stops.

Tt_{jk} = bus travel time between stop j and stop k.

The parameter t_{fix} is almost constant in practice, while Tg_{ij} and Tb_{ij} are functions of the number of passengers boarding and alighting at stops respectively. Such data is easily measurable when we know the number of passengers boarding and alighting (see sections 2.4) and the average boarding and alighting times per passenger if buses stop there.

3.1.2 The number of passengers travelling from stop j to stop k

Özekicii[12] reported that passengers arrive at a stop according to a non-stationary

Poisson distribution. In other words, the passenger behave according to a pre-specified schedule if the headway interval length is large enough. In our case, because of a high frequency of bus service (2-4 minutes on the schedule or with random delay), the number of passengers arriving at a stop can be simply treated as a Poisson distributed random variable. That is, the expected number of passengers arriving at stop j travelling to stop k during a time interval t equals to the arrival rate of such passengers multiplied by t. While the expected passenger waiting time for the next bus to arrive equals to the number of passengers waiting multiplied by t/2.

3.2 Data Collected

3.2.1 Passenger flows

Since the passenger flow volume is tremendously high in Shanghai (on a single trolley bus route 9, there are more than 50,000 passenger-trips per day), we divide the day into several time period within which we assume the passenger arrival rate is constant; then for each time period t, we make a sample survey of the flow from stop j to stop k denoted as P(t, j, k) by asking each passenger at his alighting stop, where he boarded. With this passenger flow information, we can derive:

$$\textit{the number of passengers alighting at the } k^{th} \textit{ stop during time period} = \sum_{j<k} P(t\ j\ k)$$

We can also calculate approximately, by adjusting for travel time,

$$\textit{the number of passenger arriving at stop } j \textit{ during the } t^{th} \textit{ time period} = \sum_{k>j} P(t\ j\ k)$$

An example of this data can be found in Table 1.

TABLE 1. O-D flow matrix along a route

period: 1

STOP	1	2	3	4	5	6	7	8	9	10
1	0	11	70	17	91	42	106	116	37	318
2	75	0	8	8	24	12	23	16	8	47
3	52	13	0	2	12	4	16	14	4	44
4	37	14	3	0	0	0	0	0	0	0
5	158	72	12	4	0	10	26	24	12	178
6	50	5	17	2	6	0	0	0	0	12
7	18	4	12	0	11	0	0	3	1	59
8	29	14	8	0	38	1	0	0	0	51
9	9	10	1	6	24	0	1	0	4	8
10	52	25	36	9	70	11	8	10	0	0

period: 2

STOP	1	2	3	4	5	6	7	8	9	10
1	0	21	66	10	33	24	31	132	30	253
2	75	0	11	10	25	8	9	13	5	46
3	54	13	0	6	14	2	12	10	2	42
4	37	6	13	0	0	0	0	0	0	0
5	192	26	8	12	0	8	36	32	10	146
6	45	2	6	4	9	0	2	2	4	76
7	24	3	6	0	6	0	0	4	2	50
8	18	7	5	0	11	0	1	0	0	51
9	24	4	7	1	34	1	4	0	0	42
10	68	10	23	10	54	9	14	18	3	0

period: 3

STOP	1	2	3	4	5	6	7	8	9	10
1	0	31	56	13	55	24	29	69	29	174
2	107	0	11	7	22	13	8	14	4	41
3	34	14	0	0	8	6	10	8	4	30
4	52	10	3	0	0	0	0	0	0	0
5	178	8	8	16	0	2	24	22	4	62
6	20	2	0	0	5	0	8	6	4	96
7	24	1	1	0	7	0	0	2	0	19
8	20	1	3	1	2	0	0	0	1	44
9	16	2	0	1	5	0	0	0	0	35
10	67	5	9	8	13	6	4	9	2	0

3.2.2 Bus travel time

We have access to on-board surveys of bus travel time in which a checker will write down the bus arrival time and departure times at each stop, as well as the number of passengers getting on and off the bus. The bus travel time can be obtained by subtracting the bus arrival time from its departure time at the previous stop and the fixed time due to the acceleration and deceleration. An example of a survey form is shown in Table 2.

Due to the influence of weather, seasons and holidays, those passenger flows and bus journey times may vary from day to day, so that we need a number of surveys under different circumstances which we use to adjust the data accordingly.

TABLE 2. An example of on-board survey

STOP NAME	ARRIVING TIME	PERSONS ON / OFF	LEAVING TIME	STOP NAME	ARRIVING TIME	PERSONS ON / OFF	LEAVING TIME
Wujiao			03:21:00	Hongkou			03:50:20
Guoding	03:22:00	1 / 4	22:35	Dalian	03:51:32	4 / 2	51:46
Fudan	24:23	0 / 6	24:42	Shangle	54:55	2 / 3	55:09
Dianli	24:51	0 / 14	26:05	Xinhua	56:22	2 / 5	56:37
Yunguan	26:55	10 / 3	27:10	Yixian	57:19	1 / 10	57:51
Yixian	29:01	1 / 4	29:10	Yunguan	59:32	3 / 2	59:45
Xinhua	30:41	10 / 9	30:54	Dianli	04:00:37	2 / 1	04:00:46
Shangle	33:12	10 / 8	33:32	Fudan	02:58	7 / 2	03:13
Dalian	34:50	5 / 7	35:10	Guoding	03:49	6 / 6	04:05
Hongkou	39:25			Wujiao	05:47		

From the above two surveys, we can also obtain histograms of passenger flow through the maximum load point (see Figure 1) as well as figures on the number of passengers arriving at stops during the day (see Table 3).

Figure 1. The passenger flow (person-trips) in one day through the maximum load point of the route.

TABLE 3. Number of passengers arriving at stops over time periods of a day

TIME	STOP 1	2	3	4	5	6	7	8	9	10	B2	B3	B4	B5	B6	B7	B8	B9
400	14	1	7	0	3	0	0	3	10	4	8	0	0	2	3	1	5	44
430	60	3	10	4	26	2	9	16	49	5	23	4	9	3	11	16	6	150
500	136	8	33	21	44	10	15	35	95	13	44	9	8	11	19	53	9	244
530	215	15	46	22	60	6	42	67	99	22	50	14	17	24	22	38	7	350
600	391	15	91	51	157	6	63	140	224	24	123	37	51	28	80	40	28	867
630	1036	110	204	122	265	26	72	276	374	103	301	108	73	107	131	101	56	1559
700	1652	77	305	202	457	34	71	427	483	136	222	323	162	399	176	193	69	1797
730	1933	168	286	360	394	90	156	468	590	123	233	302	69	393	97	288	49	1615
800	1171	77	118	246	186	70	136	216	300	100	86	140	28	231	66	144	14	881
830	990	64	32	141	126	54	101	77	140	109	96	50	32	190	72	111	29	822
900	626	75	57	96	99	30	44	75	129	69	59	37	38	151	55	83	7	495
930	778	91	48	104	125	88	81	94	150	85	43	29	27	128	55	58	7	637
1000	695	60	34	117	92	48	50	101	117	71	57	27	19	86	31	47	14	552
1030	644	73	49	94	80	87	44	48	142	95	34	32	14	59	55	44	14	564
1100	598	55	26	75	47	163	71	54	116	79	55	24	60	63	63	56	17	484
1130	744	79	45	119	67	83	43	51	159	70	54	32	39	95	57	58	16	637
1200	605	78	77	78	74	60	50	51	135	81	76	26	20	85	70	42	37	603
1230	869	63	70	96	110	64	107	93	237	110	109	23	38	210	74	75	27	907
1300	763	127	58	97	101	52	57	73	158	60	71	31	30	112	53	58	18	699
1330	893	66	81	131	134	97	84	146	251	75	83	51	34	104	61	112	23	863
1400	784	59	44	121	26	63	77	107	225	81	97	36	37	106	86	61	20	762
1430	712	73	39	105	100	72	91	62	317	75	82	34	44	95	76	42	21	924
1500	784	90	107	106	147	65	96	83	311	115	119	36	45	97	127	64	47	1127
1530	1004	105	95	134	176	108	105	116	721	162	97	48	56	190	140	114	54	1479
1600	917	96	132	135	180	91	103	109	577	203	126	48	238	172	247	99	39	1606
1630	884	130	197	142	358	138	87	67	620	289	107	66	62	178	117	189	30	1263
1700	1127	133	213	145	261	112	192	151	622	271	174	71	163	235	141	162	42	1655
1730	917	167	129	128	126	125	99	69	366	197	75	43	28	167	44	110	27	805
1800	1153	185	80	148	105	116	57	73	277	234	61	51	19	199	34	121	15	674
1830	597	59	43	94	82	55	47	50	201	94	65	43	29	70	41	63	18	595
1900	696	86	31	100	90	74	44	37	114	100	39	37	28	69	32	46	9	534
1930	568	63	45	88	58	62	53	49	102	97	56	23	30	93	70	45	27	489
2000	522	58	36	80	53	58	39	40	84	57	47	22	11	49	26	70	13	397
2030	817	102	64	108	108	78	35	53	91	113	35	25	13	74	43	80	14	587
2100	614	77	22	61	70	77	24	48	97	80	40	11	30	55	46	66	32	500
2130	398	38	39	61	47	29	15	44	94	58	22	32	10	51	24	57	10	289
2200	262	32	15	38	30	23	18	30	87	40	38	22	4	33	16	41	6	217
2230	212	18	13	31	36	27	15	18	93	47	13	16	27	13	32	8	198	380
2300	136	14	3	26	19	13	19	27	46	20	9	4	19	0	18	1	93	276
2330	15	2	5	4	0	1	0	0	9	3	0	1	5	0	1	0	12	212
	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

4. HEURISTIC ALGORITHMS

From the discussion in the above section, we can see it is impossible to find a solution by any exact algorithm in real time to dispatch a bus within 30 seconds, thus we have to find heuristic algorithms to solve the problem in seconds on small PC computer, which could be the only applicable solution means in a city like Shanghai.

In this section we describe three quick heuristic solution methods. The main ideas involved in first two methods are similar except for solution speed and accuracy. Method 3 uses prepared bus running patterns to dispatch buses.

4.1 Method 1

We know that the stopping variables S_{ij} are 0-1 variables and dispatching time Td_{i1} are real variables. However, in practice, the accuracy requirement for Td_{i1} is not high, so we can assume that the increase or decrease of Td_{i1} is an integer number of 30 seconds denoted as t_{unit} (i.e. t_{unit} = 30, 60,...., seconds), so that Td_{i1} becomes an integer variable. Before we can develop a procedure to solve the problem, we need to define:

T -- the vector of departure variables (i.e. T = $\{Td_{i1}\}$, i=1,...,N_b)

S -- the vector of stopping variables (i.e. S = $\{S_{ij}\}$, i=1,...,N_b, j=1,...,N_s)

s -- a solution from T and S (i.e. s = (T,S));

W(s)--the passenger waiting time for the solution s;

BEST--the objective function value of the current best feasible solution;

R_{i1} --the maximum reduction of waiting time that can be obtained from a feasible change of the i^{th} departure variable Td_{i1} in $\underline{s}$ (i.e., R_{i1} = BEST - W(s), where s is the resulting solution from the corresponding change of Td_{i1})

R'_{ij} --the reduction of waiting time from a feasible reversing the value of S_{ij} in $\underline{s}$, (i.e., $S_{ij} = 1 - S_{ij}$, and R'_{ij} = BEST - W(s))

R_{max} --the maximum reduction from a marginal change (i.e. $R_{max} = \max\{R_{i1}, R'_{ij}\}$). The corresponding solution is denoted as s^* obtained by replacing Td_{i1} or S_{ij} with Td_{i*1} or S_{i*j*} whichever results in R_{max}.

Procedure:

Generate a feasible solution $\underline{s}$ = (T,S) such as all S_{ij} = 1 and the Td_{i1} show equal headways.

Let: BEST = W($\underline{s}$)

(1) For each $Td_{ij} \varepsilon T$, **calculate** R_{i1} from a change of Td_{i1} by increasing or decreasing the departure time by a multiple of 30 seconds;
For each $S_{ij} \varepsilon S$, **calculate** R'_{ij} from reversing S_{ij} (i.e. $S_{ij} = 1 - S_{ij}$)

(2) **Calculate** $R_{max} = \max\{R_{i1}, R'_{ij}\}$, if $R_{max} < 0$, go to (3); otherwise,
Either modify Td_{i*1} or S_{i*j*} whichever resulted in R_{max}, set $\underline{s} = s^*$
BEST = W($\underline{s}$), and return back to (1);

(3) Terminate.

4.2 Method 2

This method is similar to the first one except in the way Td_{i1} and S_{ij} are updated. Method 1 only changes one of either Td_{i1} or S_{ij} in each iteration. Method 2, however, changes m variables, either Td_{i1} or S_{ij}, in each iteration. In doing so, it will speed up the solution procedure but its accuracy may be less than that of method 1. In practice, m could take a value from 2 to 5 depending on the solution speed desired by users. This method will be adopted in solving large-scale problem, that is , for the scheduling of many routes together. We can also use this method to get a good basic solution in few iterations, then switch to method 1 to further improve the results.

Procedure:

It is the same as method 1 except for step (2). Here all the feasible reductions, R_{i1} and R'_{ij}, i=1,...,N_b, j=1,...,N_s, will be calculated and sorted in nonincreasing order in a queue: $Q_1, Q_2,...,Q_l,...,Q_L$ ($L \le N_b(1+N_s)$), then set I = 0, l = 1 and start to check:

(2-a) **Let** $l = l + 1$; if $l > L$, go to (2-c);

(2-b) **If** Q_l reduction ≤ 0 return back to (2-a) to check next reduction;
otherwise: put Q_l into the list q and let I=I+1; if $I \ge m$, go to (2-c), if not

go back to (2-a);

(2-c) **If** I < m, terminate;

otherwise: change those m pair of Td_{i1} or S_{ij} whichever resulted in the reductions in the list q, and let $\underline{s} = s^*$, BEST = W($\underline{s}$), and go back to (1) as in method 1.

4.3 Method 3

The method stores a number of bus running patterns which have been most frequently used in practice. Those patterns may incorporate other restrictions coming form company policy. A solution will be selected from those prepared pattern and evaluated by the model with the estimated data.

Procedure:

Get an initial feasible solution $\underline{s} = (T,Q)$ by selecting one pattern for each bus, where $Q = \{q_i \mid q_i$ is the pattern selected for bus i$\}$, $P = \{p_k \mid p_k$ is a potential pattern$\}$

Let BEST = W($\underline{s}$).

(1) **Calculate** the feasible reduction R_{ik} = BEST - W(s) for each pair of (q_i, p_k), $q_i \varepsilon Q$, $p_k \varepsilon P$, , where s is a new solution obtained when q_i is replaced by p_k. Set $R_{max} = \max\{R_{ik}\}$, if $R_{max} < 0$, go to (3); otherwise:

(2) **Replace** q_i^* by p_k^* in Q, where p_k^* and q_i^* are the patterns resulting in R_{max}, and set $\underline{s} = s^*$, BEST = W($\underline{s}$), then return to (1);

(3) **Change** departure times by possibly increasing or decreasing a number of t_{unit} for each Td_{i1} and return to (1); or terminate if a satisfactory solution has been reached. Method 1 or method 2 could be used here to modify the Td_{i1}

5. NUMERICAL RESULTS

In this section we present numerical results by using data from actual bus operations. At time 17:06:50, three buses arrive at almost the same time at the terminus. The average headway is 2 minutes at that time. The different solutions are shown in Table 4, the total passenger waiting time if no stops were to be skipped by these three buses is 26212 person-minutes.

TABLE 4. Solutions of the problem solved in different ways.

					Skipping Pattern									
No.	Volume	Arrival	Depart	Stop:	1	2	3	4	5	6	7	8	9	10
1.	180	6.50	7.00		1	0	0	0	0	0	1	1	1	1
2.	180	7.40	8.00		1	1	0	1	0	1	1	1	1	1
3.	180	8.55	9.00		1	1	1	1	1	1	1	1	1	1

The total passenger waiting time is 21039 person-minutes.

a. Manual solution

					Skipping Pattern									
No.	Volume	Arrival	Depart	Stop:	1	2	3	4	5	6	7	8	9	10
1.	180	6.50	7.00		0	0	0	0	0	0	1	1	1	1
2.	180	7.40	8.00		0	1	1	0	0	1	1	1	1	1
3.	180	8.55	9.00		1	1	1	1	1	1	1	1	1	1

The total passenger waiting time is 20532 person-minutes.

b. Method 1 solution

					Skipping Pattern									
No.	Volume	Arrival	Depart	Stop:	1	2	3	4	5	6	7	8	9	10
1.	180	6.50	7.00		1	0	0	0	0	0	1	1	1	1
2.	180	7.40	8.00		0	1	1	0	0	1	1	1	1	1
3.	180	8.55	9.00		1	1	1	1	1	1	1	1	1	1

The total passenger waiting time is 20835 person-minutes.

c. Method 2 solution

Method 3 used a set of 25 prepared skipping patterns and the cost of each solution is evaluated by simulation. The optimizing procedure can be stopped at any time by hitting any key to get the current satisfactory solution. The best solution obtained from the method is shown in Table 5:

TABLE 5. Bus operation and its consequent results

BUS No.	ITEM	STOP									
		1	2	3	4	5	6	7	8	9	10
0	PRE-DEPARTURE (MIN)	0.00	1.82	5.77	7.49	8.61	10.71	13.04	14.64	16.61	18.31
1	PATTERN	0	0	0	0	1	0	0	0	1	1
	DEPARTURE (MIN)	7.00	8.42	11.97	13.29	14.45	16.15	18.08	19.28	21.20	22.66
	WAIT (MIN*PSN)	1008	850	694	563	432	354	242	149	68	
	REMAIN (MIN*PSN)	1008	850	694	563	146	354	242	149	0	
2	PATTERN	0	0	1	0	1	1	1	1	1	1
	DEPARTURE (MIN)	8.00	9.42	13.65	14.97	16.15	18.45	20.91	22.63	24.58	26.63
	WAIT (MIN*PSN)	1152	979	806	726	244	465	353	239	48	
	REMAIN (MIN*PSN)	1152	979	36	726	0	0	0	0	0	
3	PATTERN	1	1	1	1	1	1	1	1	1	1
	DEPARTURE (MIN)	9.0	11.31	15.32	17.45	18.70	20.84	23.16	24.81	26.74	28.74
	WAIT (MIN*PSN)	1296	1108	170	887	161	127	90	55	29	
	REMAIN (MIN*PSN)	0	538	18	737	0	0	0	0	0	

The total passenger waiting time = **20263** person-minutes.

The computer system developed also provides a graphic representation of the solution (see Figure. 2). The vertical line represent the position of each stop, the curves along the stops horizontal lines show the bus operations along the route. A circle on the curve means no stopping at that stop. The diagonal lines indicate the

number of passengers waiting at the stop over time. The heading shows the results (value of the time multiplied by passengers) from the scheduling solution. From the graph, we have a clear picture of the situation on the route.

The total passenger waiting time is 20263 person-minutes

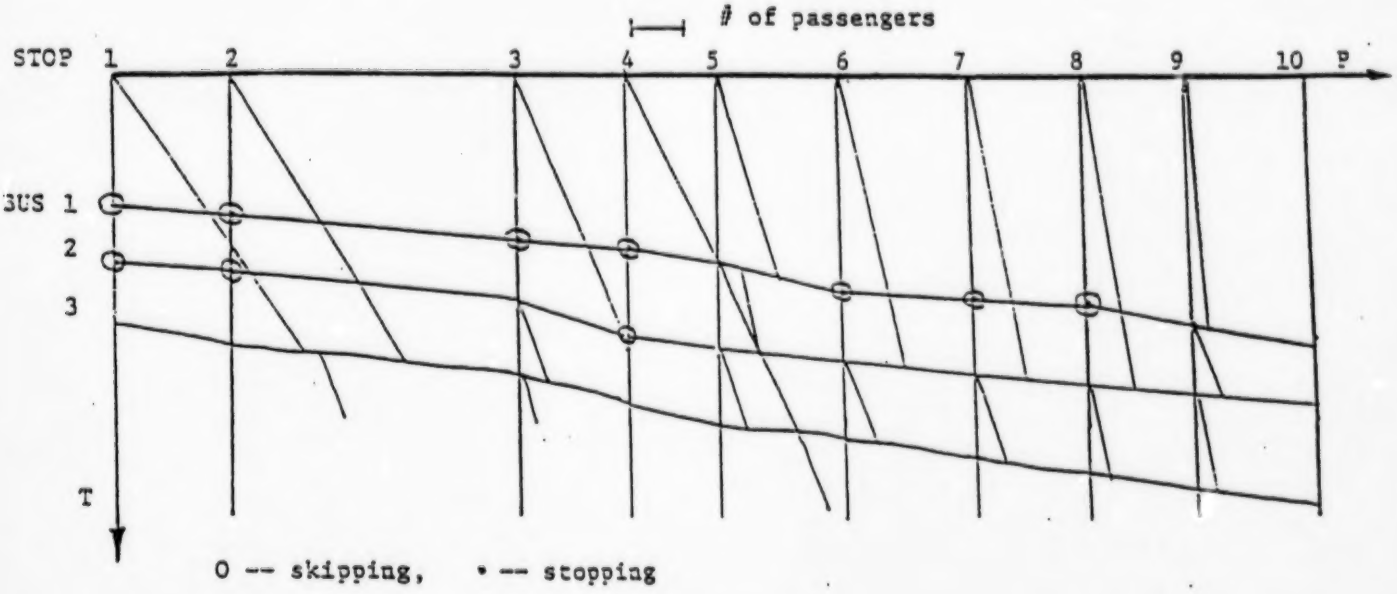


Figure 2. A graphic representation of bus operation and its consequent results

All the solutions from our system are better then manual solution.

6. CONCLUSION

We have presented a real-time bus scheduling problem and methods of solution for this problem. The method was applied to data coming from the actual operation of route 9 in Shanghai. The results of the model demonstrates the potential of the method for the improvement of transit operation in practice.

We shall explore in the future the use of the TABU search method[8] [9] in this context and to demonstrate through a full scale simulation of a bus line operation that the method provides efficient control of a bus line in real time.

REFERENCES:

[1] Bly, P.H. and R.L. Jackson, "Evaluation of bus control strategies by simulation," Transportation and Road Research Laboratory, Crowthorne, U.K. Laboratory Report **637** (1974).

[2] Bodin, et al., "Survey of routing and scheduling of vehicles and crews," Computers & Operations Research, Vol. **10**, No. **2** (1983).

[3] Breusegem, V. V. and G. Campion, "Traffic modelling and control for metro circle lines," pp 173-179.

[4] Ceder, A., "Optimal design of transit short-turn," Transportation Research Record **1221** (1989) pp 8-22.

[5] Cury, J.E., A. C. Gomide and M. J. Mendes, "A methodology for generation of optimal schedules for an underground railway system," IEEE Transactions on Automatic Control, Vol. **Ac-25**, No. **2**, (1980) pp 217-222.

[6] Daduna, J. and A. Wren (eds.), Computer Aided Transit Scheduling, Springer-Verlag, Berlin (1988).

[7] Desrochers, M.(ed.), Computer Scheduling of Public Transport 4, (this volume).

[8] Glover, F., "TABU search - Part 1," ORSA Journal on Computing, Vol. **1**, No. **3** (1989) pp 190-206.

[9] Glover, F., "TABU search - Part 2," ORSA Journal on Computing, Vol. **2**, No. **1** (1990) pp 4-32.

[10] Koffman, D., "A simulation study of alternative real time bus headway control strategies," Transportation Research Record **663** (1978) pp 70-73.

[11] Li, Y. "The application of the microcomputer in bus and crew scheduling," in: Computer Scheduling of Public Transport 2, Rousseau, J.-M. (ed.), Elsevier Science Publishers B.V. North-Holland (1985) pp 179-192.

[12] Özekicii, S., "Average waiting time in queues with scheduled batch services," Transportation Science, Vol. **21**, No. **1** (1987) pp 55-61.

[13] Rousseau, J.-M.(ed.), Computer Scheduling of Public Transport 2, Elsevier Science Publishers B.V. North-Holland (1985).

[14] Rousseau, J.-M. "Operations research in transit: a survey," Operational Research, G.K. Rand (ed.), Elsevier Science Publishers B.V. North-Holland (1988) pp 663-685.

[15] Ryan, K. T.,"Validating a bus operations simulation model," Winter Simulation Conference, San Deego, California (1979) pp 483-495.

[16] Scott, D., "Minimal fleet size in transhipment-type vehicle scheduling problems," Publication **#487**, Centre de Recherche sur les Transports, Univ. De Montreal (1986)

[17] Wren, A. (ed.), Computer Scheduling of Public Transport - Urban Passenger Vehicle and Crew Scheduling, Elsevier Science Publishers B. V. North-Holland (1981).